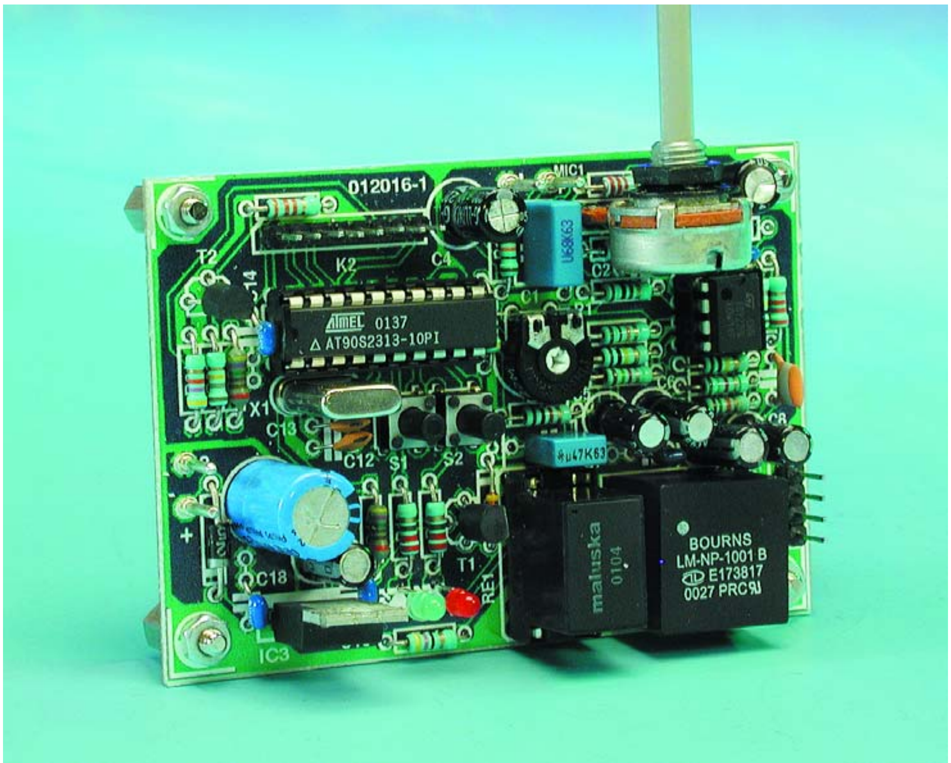
Telephone Baby Monitor

cut through the static

Design by T. Finke

The usefulness of baby monitor alarms using (FM) radio has always been limited by poor range and its susceptibility to interference. This novel design uses a microcontroller to take a fresh look at the problem.



There are several different types of baby monitor on the market. The most common provides a short-range radio link (typically 100 m) between the transmitter unit in the nursery and the receiver unit carried by a parent. Another type uses audio to modulate a carrier frequency sent along mains wiring to the parents receiver plugged into a mains outlet. The RF model has a limited range and the mains modulation method requires that the parent cannot wander too far from a mains socket. Both systems are susceptible to interference.

The device described here goes about it in a different way: The unit is plugged into the house telephone socket and positioned in the nursery so that its in-built microphone can pick up sounds that the baby makes. The baby monitor microcontroller will analyse all sounds and determine if it hears a cry. When a cry is detected and persists for a given time, it will dial-out a pre-programmed number (typically the parents' mobile). There is no need to answer the call, a glimpse at the mobile's display will indicate that the house phone called you and junior is demanding attention!

Noise discrimination

The baby monitor circuit diagram is shown in **Figure 1**. The electret microphone MIC1 detects sounds in the nursery. Resistor R1 provides a bias from the supply rail while R7 and C4 are used to decouple any noise on the microphones supply. The AF (audio frequency) output signal is fed to the inverting input of IC2.A via capacitor C1. IC2 is a so-called rail-to-rail dual op-amp from ST (available from Farnell). This type of IC is necessary in this application because it allows us to set a high

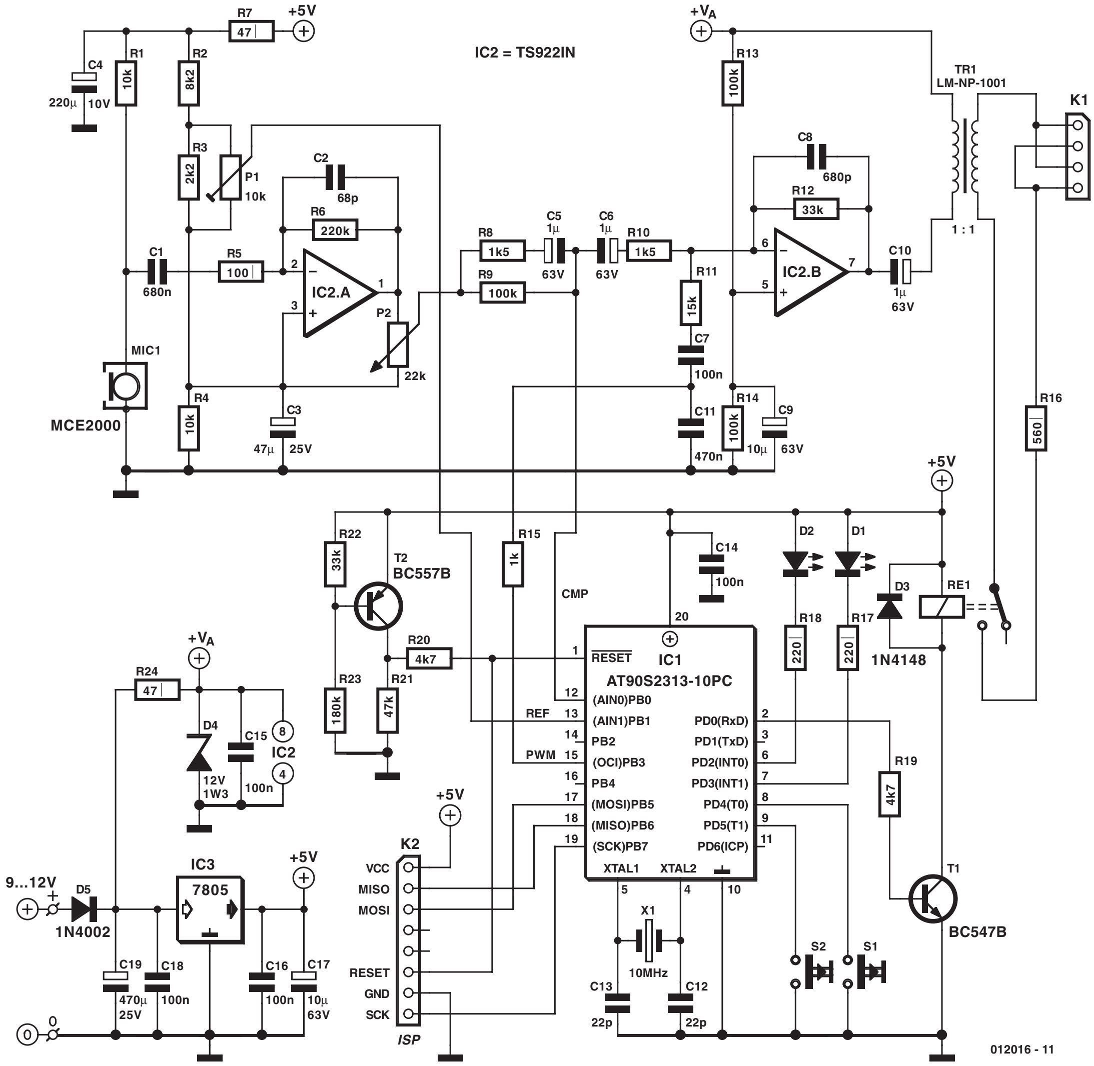


Figure 1. The circuit contains both analogue and digital circuitry.

level of amplification. The supply voltage to this IC is 12 V.

The audio signal is output from this stage on the wiper of preset P2. The signal quality is not especially hi-fi but it is not necessary for this application. Any high frequency ringing or oscillations are damped by capacitor C2.

The potential divider formed by resistors R2, R3, P1 and R4 produces a fixed voltage at approximately half-rail (2.5 V) for the non-inverting input of IC2.A capacitor C3 smoothes out this voltage.

The signal is now fed to the comparator input of IC1. **Figure 2** shows the internal features of this chip. One input of the comparator (AIN1) is connected to an adjustable reference voltage (REF). The other input (AIN0) is connected to the AF signal (CMP) and when this signal rises above the reference voltage the comparator output will switch and generate an interrupt in the microcontroller. If the microcontroller establishes that the sound is a baby crying it will change the AIN0 pin to an output and then output a zero to effectively short out the AF signal (R8 limits the short circuit current). Next the microcontroller sends out the DTMF signals on the PWM pin to dial the number. A more detailed description of this process is described under the heading 'DTMF signals'.

Amplifier IC2.B has a dual purpose to amplify audio or the PWM signal. The amplification factor required here is around two for DTMF signals and ten for audio signals. Resistor chain R13 and R14 provide a reference voltage of half V_A for the non-inverting input of IC2.B. Capacitor C9 smoothes any noise on this reference voltage. Both amplifiers have a bandwidth of 10 kHz, which is more than enough for this application.

Brown-Out protection

Brown-outs occur when the mains supply undergoes a brief interruption, just enough to cause your houselights to dim momentarily. These interruptions cause havoc in microcontroller systems that do not have brown-out protection. The microcontroller supply voltage starts to fall, producing unreliable operation but then recovers again before a reset can be generated. The processor can mis-read data values and instructions, ending up in an undefined state and systems using EEPROMs can have stored values corrupted. The brown-out protection circuit used here is built around transistor T2 and is taken from an ATMEL application note. This configuration is more economical than commercial brown-out protection IC's. The circuit quickly generates a RESET pulse when it detects a dip in the supply voltage.

Telephone connection

A length of telephone cable terminated with a line plug is suitable for connecting the baby monitor to a UK telephone socket. The type 431A connector is the UK standard line plug. The plug moulding is six-way but contacts are only fitted to the inner four positions numbered 2, 3, 4 and 5. Pins 2 and 5 should be connected to 'a' and 'b' on connector K1. The polarity is unimportant.

When the baby monitor dials out, transistor T1 conducts and the contacts of RE1 close, loop current is limited by R16 and the secondary winding of TR1. DTMF dialling tones are now output to the primary side of TR1 from IC2.B.

User interface

The baby monitor user interface consists of pushbuttons S1 and S2 to control the unit and enter data while LEDs D1 and D2 display its status. This interface is not the most elegant but you will seldom need to change system parameters so the additional expense of a dedicated keypad and LCD can hardly be justified in this instance.

A mains adapter unit supplying 9 to 12 V provides power for the baby monitor. A protection diode (D5) ensures that reversing the power supply leads cannot damage the input. C19 acts as a smoothing capacitor to reduce any supply ripple. IC3 provides 5 V for the circuit while the opamp supply is taken directly from the unregulated input from the mains unit. Diode D4 ensures that this supply does not exceed the 12 V maximum for the opamp, its purpose is to protect against the higher no-load voltage output from unregulated mains units. Do not expect it to be able to cope with a continuously higher input from a mains unit rated at more than 12 V.

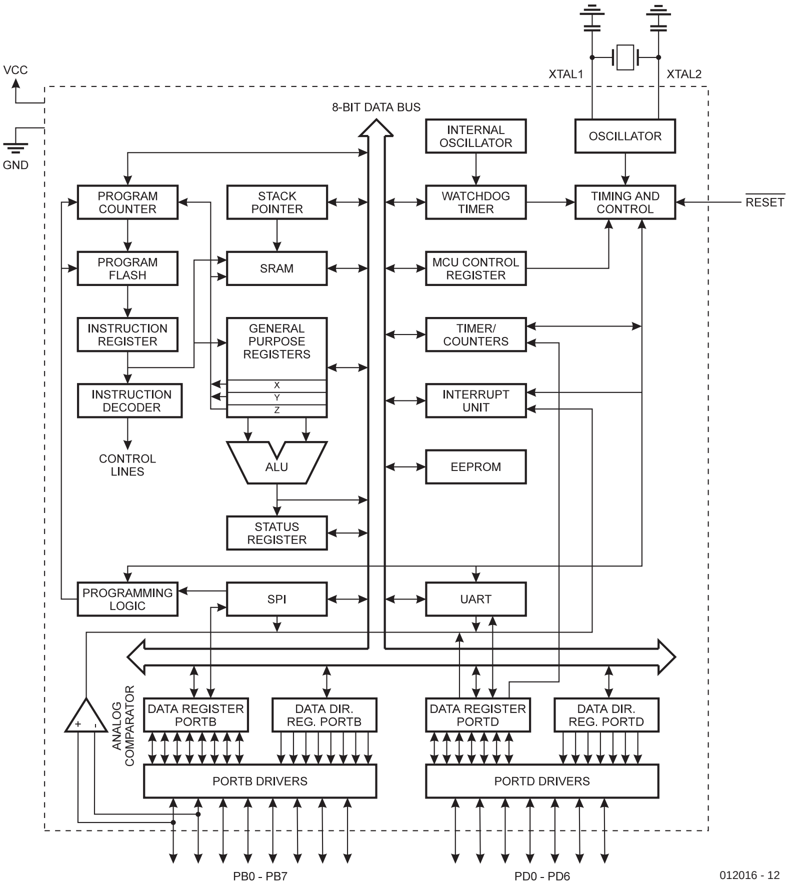


Figure 2. The controller uses an on-board comparator.

It would be possible to power the unit directly from the telephone network but such practices are frowned upon by telephone authorities and besides on safety grounds it is better to maintain a good galvanic separation (no dc path) between the relatively high voltage of the phone network and the baby monitor unit.

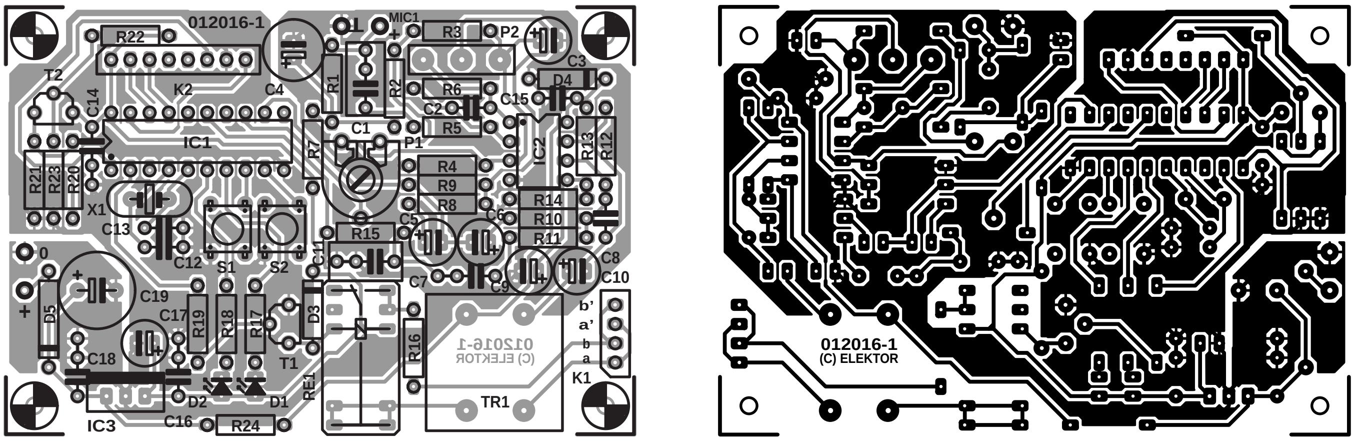


Figure 3.Single-sided PCB layout for the baby monitor.

The microcontroller's clock frequency of 10 MHz may seem high but it is necessary to achieve the accuracy for DTMF tone generation. The microcontroller can be programmed via connector K2. The pin-out of this connector corresponds to the standard used by Lattice to program their ispCPLDs. The same cable can be used to download the program to Atmel as well as Lattice chips. If you plan to order a preprogrammed controller from the Publishers (and have no need to re-program it) then you can omit connector K2.

Assembling and adjusting

The compact single-sided PCB is shown in **Figure 3.** Fit all the components to the board starting with the small items and finishing with the largest. There are no wire links necessary with this layout. For the pushbuttons S1 and S2 choose types with long buttons so that they protrude through the plastic case for easier access.

The microphone insert should be mounted under the case for protection but don't forget to make an opening in the case so that sound can reach the microphone. The telephone cable outlet from the case should be fitted with a strain relief exit grommet.

A final check of the PCB is always a worthwhile activity before the circuit is powered up. Make sure all components are correctly positioned and that there are no solder bridges between the PCB tracks. At power-up the green LED indicates that the unit is operational. If a sound is detected, this LED will start to flash. Pressing pushbutton S1 will turn off the unit and the green LED will go out. Preset P2 is used to adjust the sound level so that the cry from the

COMPONENTS LIST

Resistors:
R1,R4 = 10kΩ
R2 = 8kΩ2
R3 = 2kΩ2
R5 = 100Ω
R6 = 220kΩ
R7,R24 = 47Ω
R8,R10 = 1kΩ5
R9,R13,R14 = 100kΩ
R11 = 15kΩ
R12,R22 = 33kΩ
R15 = 1kΩ
R16 = 560Ω
R17,R18 = 220Ω
R19,R20 = 4kΩ7
R21 = 47kΩ
R23 = 180kΩ
P1 = 10kΩ preset H
P2 = 22kΩ potentiometer, linear, mono, miniature version

Capacitors:
C1 = 680nF
C2 = 68pF
C3 = 47µF 25V radial
C4 = 220µF 10V radial
C5,C6,C10 = 1µF 63V radial
C7,C14,C15,C16,C18 = 100nF ceramic
C8 = 470pF
C9,C17 = 10µF 63V radial
C11 = 470nF
C12,C13 = 22pF
C19 = 470µF 25V radial

Semiconductors:
D1 = LED, red, 3mm
D2 = LED, green, 3mm
D3 = 1N4148
D4 = zener diode 12V 1.3W
D5 = 1N4002
T1 = BC547B
T2 = BC557B
IC1 = AT90S2313-10PC, programmed, order code **012016-41**
IC2 = TS922IN (Farnell)
IC3 = 7805

Miscellaneous:
K1 = 4-ay SIL header
K2 = 8-way SIL header
S1,S2 = pushbutton, 6x6 mm
Re1 = subminiature relay 16x119x11.5 mm with SPDT contact, e.g., Maluska FRS1B-S DC 5V, (5 V, 56 Ω, Conrad Electronics # 505188)
X1 = 10MHz quartz crystal (C_{load} = 32pF, parallel resonance)
Tr1 = line transformer, Bourns LM-NP-1001 B)
MIC1 = electret Microphone element (e.g., Monacor/Monarch MCE2000)
PCB, order code **012016-1**
Disk, C source code and hex files, order code **012016-11**
(see Readers Service page)

DTMF Signal Generation

Simple DTMF generator IC's (like the TP5088 from NS) are practically no longer available. These days DTMF generators come with all sorts of bells and whistles that we do not need in this application. They are generally difficult to find and have lots of interface signals to take care of (The MT80444 needs the Φ2-Signal from the obsolete 6502 processor). A good solution to this problem is to generate these tones in software.

DTMF stands for Dual Tone Multi Frequency signalling (tone dialling). Eight frequencies are used comprising a group of 4 low tones and a group of 4 high tones. Each of the 16 possible keys on a telephone keypad is represented by a different combination of one high tone and one low tone. The reasoning behind this was that it is unlikely that these sounds would occur in normal speech patterns. The frequency of all the eight tones needs to be accurate to within 1.5 % otherwise the exchange equipment may not recognise the DTMF signal as a key press.

A D/A converter would be an ideal component to generate the tones necessary for DTMF signalling. Unfortunately the Atmel microcontroller used in this project does not have such a device on-board. The controller is however equipped with a Pulse Width Modulated (PWM) output and this can do the job just as well. This output has a resolution of 8 bits with a clock frequency of 19.6 kHz (fCK/510). The dual sine waves are produced by loading the PWM counter with values from a look-up table to control the width of the pulse. The spectrum of the resultant waveform shows peaks at the two sine frequency fundamentals and a peak by f = 0 (this dc component is removed by capacitors C7 and C11). Other frequency components of this signal are at the PWM switching frequency. The low-pass filter formed by R15 and C11 ensures that these square wave components on the output signal are removed to leave just the two sine waves. The corresponding amplitude components for the high and low tone groups are stored in a sine wave table. If the stored period does not result in the necessary frequency accuracy then two or three different values are used. Altogether the sine wave table contains approximately 200 bytes of data.

baby does not overload the input to IC2.B. This setting is best performed by viewing the output of IC2.B with an oscilloscope while adjusting P2. Lastly, the monitor trigger level is adjusted by preset P1. This adjustment ensures that the unit does not respond to background noises in the nursery. The baby monitor will not dial-out each time this comparator is triggered, the frequency of the sound must be above a threshold, and the sound must persist for a time greater than a minimum period. Both of these parameters are stored in EEPROM and can be altered at will.

When the microcontroller recognises the sound as a baby cry it will switch relay RE1 to connect the monitor to the telephone line (equivalent to lifting the receiver of a phone), the red LED will switch on continuously and then it will call the stored telephone number by using tone dialling. After a preset ring time (two to three rings) the monitor will hang up. The monitor will now remain in a 'standby' state for a pre-programmed period to prevent it from being continually triggered. The red LED will flash during this period, finally the unit will switch back to its operational mode.

Dial out

Pressing pushbutton S2 will immediately call the dial-out phone number stored in memory. External equipment (e.g. house alarm or temperature monitor) with a normally open type relay or open collector output can wire this output in parallel to S2 so that the baby monitor will dial out when the alarm condition is detected.

Enter the number

The method of entering the dial-out phone number is a little long winded but once it is stored in EEPROM the chances are that you will not need to change it too often. A long press of pushbutton S1 (longer than two seconds) will switch the monitor unit into programming mode (the red LED will come on and the green will go out). Now enter the first digit of the dial-out phone number: press pushbutton S1 corresponding to the digit (if the number is zero do not press S1 at all) after tapping in the digit press S2 to enter it and move on to the next digit of the number. Carry on until the complete number is entered and then a long press of S1 will finish the process. If you need to program a time delay in the dialling process (The number may involve a connection through an exchange) then just press S1 10 times where you want the delay in the phone number. If you make a mistake when entering a digit then two long presses of S1 will return you to the start of the number and you can make another attempt at re-entering it.

Checking the number

The stored dial-out phone number can be checked by pressing and holding pushbutton S2. The red LED will now light continuously and the green LED flashes to indicate the value of the first programmed digit, repeating after a short delay (the LED will remain off if the number is zero). Pressing S2 then moves on to the next digit. After the last digit the red LED will go out and the green LED will light continuously to indicate that the unit is ready for use.

Programming

All the baby monitor software is written in 'C'. Constants used in the program (the noise detector minimum frequency, various timer values and the dial-out phone number) may be changed without resorting to Compiler software but by using a simple programming device to directly overwrite the values stored in EEPROM:

Address	Function	Default
\$00-\$1F	Dial-out phone No. (ASCII coded). \$00 is the terminating character.	
\$20	Low frequency cut-off. Sounds above this frequency are classified as a baby cry. ($f_g = X \cdot 38$ Hz)	10
\$21	Length of the monitoring window after sound detected $2^X \cdot 6.5$ ms	10
\$22	The number of 'noisy' time periods (of 6.5 ms) in the monitoring window before the alarm is triggered.	80
\$23	Time between dialling and hanging up (ring time). $X \cdot 6$ s (approx.)	4
\$24	The baby monitor is in standby mode after dialling-out. It will wait for this period of time ($X \cdot 16$ s) before listening for noise again.	10
Where 'X' is the value stored in memory		

Pressing S1 and S2 together at power-up will load the default values to EEPROM.

(012016-1)